Understanding the Global Attack Toolkit Using a Database of Dependent Classifiers*

Peter Mell
NIST, Computer Security Division
Peter.mell@nist.gov
11/5/98

Abstract

High profile Internet web sites publish a large collection of attack scripts that we call the Global Attack Toolkit (GAT). It is a dangerous tool available to the average web surfer and yet we know little about this set of attacks besides the fact that it exists. We have taken a sample of 119 attacks from the GAT that were published between May and October 1998. We classify these samples with dependent classifications and store the results in a database. Using the database, we generate statistics on important characteristics of the GAT (e.g. what percentage of attacks are launchable from a Windows host, what percentage are remote penetration attacks, and what percentage use UDP).

One can also use the database as a forensic tool and as an attack script search tool. As a forensic tool, a search on the database creates a list of attacks that could have compromised a penetrated system. As an attack script search tool, similar search techniques yield lists of attacks that conform to desired specification.

Background and Introduction

For many years security professionals, especially intrusion detection specialists, have yearned for an attack database for a variety of reasons, such as to test and improve their systems or products. Some have postulated that a certain unnamed Government agency has a secret exploit script collection which, if released, would be a silver bullet for security researchers attack script collection efforts. In my previous role as an academic researcher, my colleagues and I spent much time attempting to obtain an attack database from the military, only to find that it contained just 18 attacks. In another effort, the Defense Advanced Research Project Agency (DARPA) funded a multi-million dollar intrusion detection and response project which used only 11 attacks for testing. Finally, another million-dollar DARPA project to test intrusion detection systems used 30 attacks.

These numbers seem to indicate that obtaining attacks is difficult. However, this is not true. The Rootshell web site has a database of 690 exploit scripts† [1]. Fyodor's Playhouse contains 383 attacks [2]and the Legacy hacking site has 556 exploits [3]. The disparities in numbers between what the security experts use and what is available lies in the fact that most attack scripts are not general purpose and require a very specific target. They generally require application X version Y running on operating system Z. This breadth of targets makes it difficult for security professionals to collect a large number of attacks that they can run and demonstrate.

However, this fact does not diminish the importance of the attacks available on the Internet. Together, the exploit script web sites form an attack toolkit that is available to literally everyone in the world. The enormous number of attack-target types limit the security professional in his work, but does not limit the hacker. Somewhere on the Internet there exists a host vulnerable to almost every attack, and scanning tools are readily available to find that host.

---

* Published in the proceedings of the 2nd Workshop on Research with Security Vulnerability Databases, January 21-22, 1999 at Purdue University.

† A small percentage of these "exploits" are really vulnerability descriptions, tools to cover an attacker's tracks, or security tools.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	11/5/98	Report

4. TITLE AND SUBTITLE
Understanding the Global Attack Toolkit-Using a Database of Dependent Classifiers

5. FUNDING NUMBERS

6. AUTHOR(S)
Peter Mell

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
IATAC
Information Assurance Technology Analysis Center
3190 Fairview Park Drive
Falls Church VA 22042

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Defense Technical Information Center
DTIC-IA
8725 John J. Kingman Rd, Suite 944
Ft. Belvoir, VA 22060

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

12b. DISTRIBUTION CODE
A

13. ABSTRACT *(Maximum 200 Words)*
High profile Internet web sites publish a large collection of attack scripts that we call the Global Attack Toolkit (GAT). It is a dangerous tool available to the average web surfer and yet we know little about this set of attacks besides the fact that it exists. We have taken a sample of 119 attacks from the GAT that were published between May and October 1998. We classify these samples with dependent classifications and store the results in a database. Using the database, we generate statistics on important characteristics of the GAT (e.g. what percentage of attacks are launchable from a Windows host, what percentage are remote penetration attacks, and what percentage use UDP).
One can also use the database as a forensic tool and as an attack script search tool. As a forensic tool, a search on the database creates a list of attacks that could have compromised a penetrated system. As an attack script search tool, similar search techniques yield lists of attacks that conform to desired specification.

14. SUBJECT TERMS
GAT, Security, Attacks

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	UNCLASSIFIED	UNCLASSIFIED	None

We call the set of attacks available on hacker/security web sites the Global Attack Toolkit (GAT). It is a resource used by security professionals to test their systems and security software. Without it, security experts would be blind to available attacks while the pirates would continue underground dissemination. However, the GAT is also available to teenagers, hackers, and the military of our world's nations. It may become a major weapon in the information wars of the future. It is not known how many people access the GAT, but Rootshell alone has 27,300 people on its mailing list [4]. While the GAT has the potential to be extremely dangerous, researchers are doing little work to understand the nature of the GAT. What are its characteristics? How has it changed over time? More importantly, can we detect future trends? And what level of sophistication is required to use the GAT?

We propose to answer these questions by creating a database of attacks indexed by a set of dependent classifiers. The attacks in the database will come from a sample of the GAT. By statistically analyzing the distribution of the attacks over time we hope to gain an understanding about the nature of the GAT and how the GAT is changing.

Past Efforts

The classification of vulnerabilities has received much attention in the past [5 6 7 8 9]. However, this work does not directly carry over into the area of classifying attacks. Before discussing vulnerability and attack classification schemes, we define our terminology:

Vulnerability: A misconfigured or faulty element of a computer system that can be exploited for unauthorized use of the computer.

Attack: A script or set of instructions that an attacker could use to do something unauthorized to a host or set of hosts.

Incident: An event when an attacker uses an attack against a target.

Vulnerability classification work does not directly apply to attack classification work because attacks have features that do not exist in vulnerabilities:

- Goals,
- Specifications for the attacking host,
- The capability to exploit multiple vulnerabilities,
- Transmission methods by which the attacker reaches the target, and
- Requirements the attacker must meet in order to launch the attack.

In addition, unlike vulnerability classification schemes, attack classification schemes are not necessarily concerned with identifying the specific exploited flaw.

Another difference between vulnerability classifications and attack classifications is that vulnerability classification schemes strive to use independent classifiers while attack classification schemes must rely on dependent classifiers. Dependant classifiers allow one to choose multiple categories within a class while independent classifiers force one to choose a single category for each class. Using our definition of an attack, there exists an attack that no attack classification scheme that uses independent classifiers can uniquely classify. The reason is that our definition of an attack does not specify how many vulnerabilities the attack can exploit, how many goals the attack may achieve, or how many transmission methods it can use. No reasonable scheme of independent classifiers will uniquely classify an attack that is the union of all existing attacks.

John Howard defines an attack as "a single unauthorized access attempt, or unauthorized use attempt, regardless of success" and an incident as "a group of attackers, and the degree of similarity of sites, techniques, and timing" [10]. We view Howard's 'attacks' and 'incidents' as incidents which makes his attack classification work not applicable in this domain. Howard's definition of an 'attack' is narrower than ours and thus may seem more useful. However, the GAT

contains many scripts that have multiple goals, multiple transmission methods, work against multiple targets, from multiple platforms, and that exercise multiple vulnerabilities. Scripts in the GAT force us into a definition of "attack" that makes classifying with independent classifiers difficult.

Our Approach

Our approach is to use a classification scheme that is fine grained enough to yield interesting results but broad enough to allow us to quickly characterize attacks. It takes on average 5 minutes to classify an attack, which means that we can potentially characterize thousands of attacks.

We classify each attack within the following major categories: script goals, target types, attacker platforms, transmission methods, attacker requirements, targets of attack, and if it counts as an attack toolkit. In addition, we record the script name, publication date, source web site, and a description of the attack. Figure 1 shows the classifiers we use.

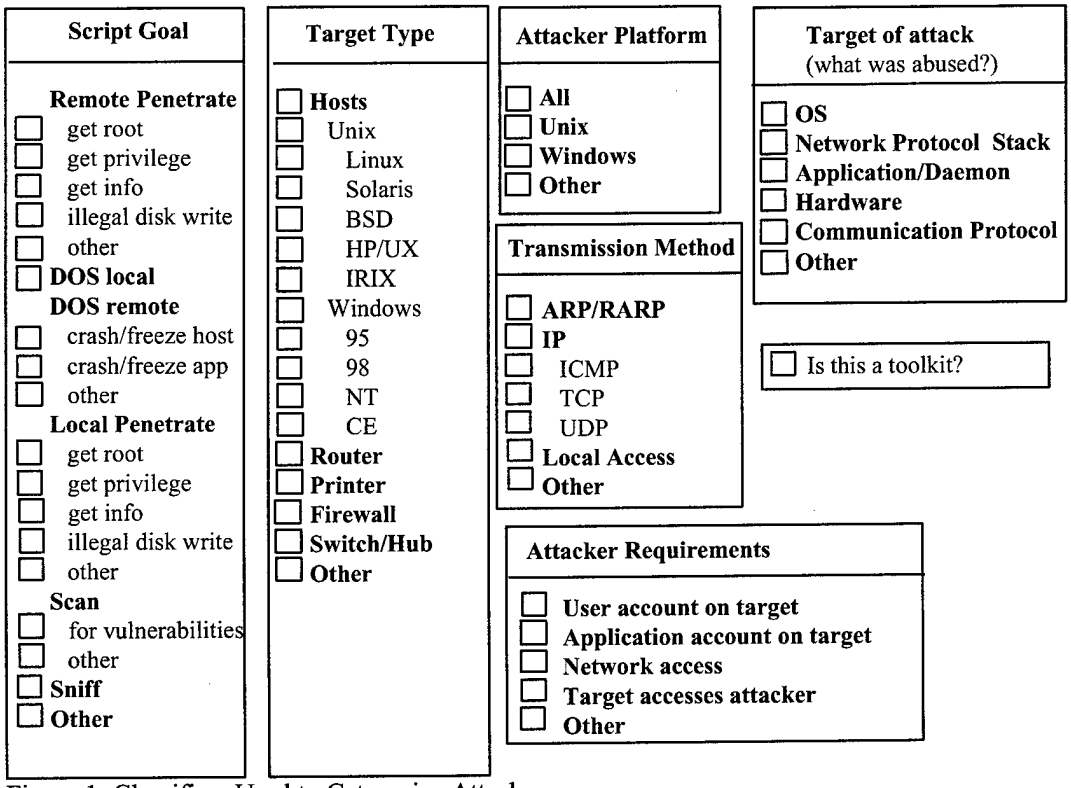

Figure 1: Classifiers Used to Categorize Attacks

Most categories are self-explanatory, but a few explanations are required:

1. We define a "toolkit" as a single exploit script that exercises more than one vulnerability.
2. Classes that are higher in a hierarchy can mean "other", "all", or a subset of the lower classes. For example, checking the target type Unix could mean that the attack effects all Unix hosts, a subset of the Unix operating systems listed, or a completely different Unix operating system not listed. As another example, we mark IP when we are uncertain which protocol the attack uses.

3. Under "attacker requirements", the "application account on target" category means that the attacker has an account with some application residing on the target.
4. Under "attacker requirements", the entry "target accesses attacker" is for any attack that requires the target of the attack to access the attacker's information in order for the attacker to launch the attack. This is most common with web sites that attack users that visit.
5. Under "target of attack", we mark that part of the machine that the attack abuses. Often, it is difficult to determine whether a daemon is an application or part of the OS. Our solution is to consider all daemons to be applications. Since we also considered the network protocol stack to be separate from the OS, this means that usually only local attacks can abuse the OS. If a communication protocol specification was not followed (e.g. fields set too large) in order to attack, the attack abused the network stack. If the attack used the network protocol correctly, but took advantage of flaws in the protocol specification, then we mark the "communication protocol" entry.

Our initial effort has focused on categorizing the attacks available on Rootshell. They were chosen because of their reputation for being an up to date resource, the number of attacks they offer, and the fact that they record a publish date for each attack. We have entered 119 attacks into our database that covers the attacks published from May to October 1998.

We record the classification results in a Microsoft Access 97 database to facilitate data analysis. Besides obtaining statistics on the characteristics of GAT, alternate uses for the database are identifying possible attacks that caused a host to become penetrated and aiding researchers in finding attacks of particular types.

When a host is penetrated, it is often not obvious how the penetration occurred. However, our database can be used as a forensic tool to identify what attacks could have penetrated the system. Searching the database with the characteristics known about the host and attack to create a list of attack candidates does this. Of prime importance in this application is that we record a description of the attack in the database. The security expert can search for records that have key words in the description that correspond to components that exist on the penetrated host. For example, Rootshell was recently penetrated and defaced [11]. It was running Linux 2.0.35, ssh 1.2.26, qmail 1.03, Apache 1.3.3, and nothing else on its network interfaces. We can use these keywords to search the attack descriptions while listing only attacks that are in the following categories provides additional filtering:

1. Script Goal: Any of the remote attacks
2. Target Type: Hosts or Unix or Linux
3. Transmission Method: IP or ICMP or UDP or TCP or Other
4. Attacker Requirements: network access or target accesses attacker or other

The SQL statement would AND together requirements 1 through 4 and then AND that result with the results obtained from searching the attack description for keywords. The result will be a list of candidate attacks that could penetrate the host.

Another use of the database is to create a list of attacks that will work in a particular environment. A security professional wanting to attempt to penetrate a target can use the classifiers and the attack description to narrow down his scope of search in a similar way as presented above.

Results: Statistics on the GAT

While we can use the database to search for attacks, our real purpose is to use it to discover the characteristics of the GAT. To this end, we took statistics on the attacks published by Rootshell for six months from May until October of 1998 totaling 119 attacks. Since the period over which we sampled the GAT is small, we will not attempt to discuss trends in the GAT over time. However, we can present characteristics of the GAT as it exists in the present:

General:
The percentage of attacks that:
Are launchable from Windows (37%)
Are launchable from Unix (95%)
Are a toolkit (3%)

Target of Attack: (What part of a network element did the attack abused?)
The percentage of attacks that:
Attack an application (61%)
Attack an OS, hardware, or communications protocol (27%)

Script Goal:
The percentage of attacks that:
Are penetration exploits (53%)
Are remote penetration exploits (28%)
Are remote get root or privilege exploits (19%)
Are local penetration exploits (26%)
Are local get root or privilege exploits (19%)
Are local denial of service exploits (4%)
Are remote denial of service exploits (27%)
Are scanning tools (13%)
Are scanning tools that look for vulnerabilities (8%)
Are network sniffers (2%)

Transmission Method:
The percentage of attacks that:
Use TCP (at least 41%, at most 58%)*
Use UDP (at least 8%, at most 25%)
Use ICMP (at least 3%, at most 19%)
Use a protocol other than IP (3%)

Target Type: (Kind of host or network element)
The percentage of attacks that:
Work against hosts (91%)
Work against Unix hosts (at least 34%, at most 73%)†
Work against windows hosts (at least 18%, are most 56%)
Work against network elements other than hosts (13%)
Work against routers (6%)
Work against printers (7%)
Work against firewalls (7%)
Work against switches or network hubs (2%)
Work against switches or network hubs and are remote penetration attacks (2%)

Attacker Requirements:
The percentage of attacks that:
Require the target to access the attacker's site/application (4%)

---

* We write these results as ranges because we could not quickly identify which IP protocol some attacks used.

† We write these results as ranges because we could not quickly identify the hosts against which all attacks would work.

These results represent attack scripts available to the public. We find it interesting that Windows machines can launch 37% of attacks. It is simply not true that one needs a Unix box to be dangerous. Another interesting result is that 28% of newly published attacks are remote penetration exploits. It was the author's understanding that remote exploits were rare but that is not the case. Remote penetration exploits exist that penetrate even network switches which we expect to have a high level of security.

Do not be alarmed at the high percentage of remote denial of service attacks, 27%. Many denial of service attacks are impossible to prevent which means that as software developer's patch vulnerabilities, hackers will write only denial of service scripts. Unfortunately, that day is not now and vulnerabilities leading to penetration scripts abound.

Many security experts fear that attack toolkits will in the future be posted on the Internet to automatically scan sites for vulnerable hosts and then exploit them at the push of a button. Even at the annual hacker DEF CON conference, hackers expressed concern over this type of development [12] [13]. We hope that, for the sake of our nation's security, these toolkits are not published. Currently, only 3% of attacks exploit more than one vulnerability. The most dangerous toolkit contained 30 exploits neatly bundled with a GUI. And this tool could easily be combined with excellent vulnerability scanning tools like nmap and queso to produce an attack that at the push of a button can automatically acquisition targets on the Internet and penetrate them [14] [15] [16]. For now, however, hacking still takes some knowledge, skill, and patience.

Some hackers will need patience as new attacks have come out that require a target to visit the hacker's web site or host. Four percent of new attacks have this characteristic. The Internet may begin to develop "bad parts of town" analogous to many inner cities. Watch where you walk!

Future Work

We will continue to monitor the number of toolkits that are published and will attempt to create characteristics that will measure the sophistication and threat created by any particular toolkit.

We will research trends in the GAT over time. A major focus will be to create new characteristics that will measure interesting features of the GAT over time. An advantage of using dependent classifiers is that we can add a characteristic to our database without changing the classification scheme.

It will be interesting to take data from multiple web sites and to see if the statistics yielded from site to site are consistent. If they are not, then the results published in this paper are the result of the bias of Rootshell and not a true sampling of the GAT.

Conclusion and Summary

The GAT is a tool that informs software developers and security experts about available attacks and thus enables them to secure their sites. The GAT is also a widely available and dangerous tool. Simply typing "hacking" into a major search engine will produce references to sites with exploits. Many of the search engines (Excite, Lycos, Infoseek, and Yahoo) have a predefined subject area for hacking sites that contain hundreds of exploits [17] [18] [19] [20]. It is interesting that the best sites, by subjective analysis, have almost the same exploits. This means that the same attack scripts are easily available to anyone who can type the word "hacking". The GAT is becoming more of a reality ever day.

Our database of attacks has proved valuable in characterizing the nature of the GAT. We revealed the proportions of attacks that have specific goals, transmission methods, target types, and launching platforms. The statistics revealed that teenagers with little knowledge and windows machines are able to launch 37% of the attacks. The statistics revealed that while attack toolkits are still not popular they do exist and coupled with the 7% of attacks that scan for vulnerabilities

they could become very dangerous. They revealed that firewalls, routers, and network hubs are not resistant to remote attacks. And the statistics revealed that 4% of attacks involve the target wandering into an attacker's trap which means that people may not be able to web surf with impunity.

References

[1] Rootshell, http://www.rootshell.com
[2] Fyodor's Playhouse, http://www.insecure.org
[3] Legacy hacker site, http://www.jabukie.com/The_Legacy_Main_Page.htm
[4] Rootshell, http://www.rootshell.com
[5] Taimur Aslam, Ivan Krsul, and Eugene Spafford. Use of a Taxonomy of Security Faults. *19th National Information Systems Security Conference*, October 22-25, 1996 Baltimore, Maryland
[6] Jim Carlstead, Richard Bibsey II, and Gerald Popek. *Pattern-directed protection evaluation*. Technical report, Information Sciences Institute, University of Southern California, June 1975.
[7] R.P. Abbott et al. *Security Analysis and Enhancements of Computer Operating Systems*. Technical Report NBSIR 76-1041, Institute for Computer Science and Technology, National Bureau of Standards, 1976
[8] Carl Landwher et al. *A taxonomy of computer program security flaws*. Technical report, Naval Research Laboratory, November 1993.
[9] Brian Marick. A survey of software fault surveys. Technical Report UIUCDCS-R-90-1651, University of Illinois at Urbana-Champaign, December 1990
[10] John Howard. *An Analysis of Security Incidents on the Internet 1989-1995*, Carnegie Mellon University, April 1997
[11] Rootshell, http://www.rootshell.com
[12] Joel Deane. *Fear and hacking in Las Vegas*, http://www.jabukie.com/news.html, November 1998
[13] DEF CON Conference, http://www.defcon.org
[14] Stuart McClure, Joel Scambray. TCP fingerprinting solutions for Linux offer another way to gather security data, Infoworld, October 26, 1998, http://www.infoworld.com/cgi-bin/displayNew.pl?/security/981026sw.htm,
[15] nmap scanning tool, http://www.insecure.org/nmap
[16] queso scanning tool, http://www.apostols.org/projectz/queso
[17] Excite search engine, http://www.excite.com, Directory listing: "Computers and Internet::Web Site Guide::Programming::Hacking", November 1998
[18] Lycos search engine, http://www.lycos.com, Category listing: "Computers > Software > Programming > Hacking" and "Entertainment > Just For Fun > Trends > High Tech > Computer Hacking", November 1998
[19] Infoseek search engine, http://www.infoseek.com, Reviewed web site topic: "Internet>Net community>Internet politics>Cybercrime>Hacking", November 1998
[20] Yahoo search engine, http://www.yahoo.com, Category listing: "Computers and Internet: Security and Encryption: Hacking", November 1998